STAT 4360 (Introduction to Statistical Learning, Fall 2022)
Mini Project 6
Name: Haeun Kim

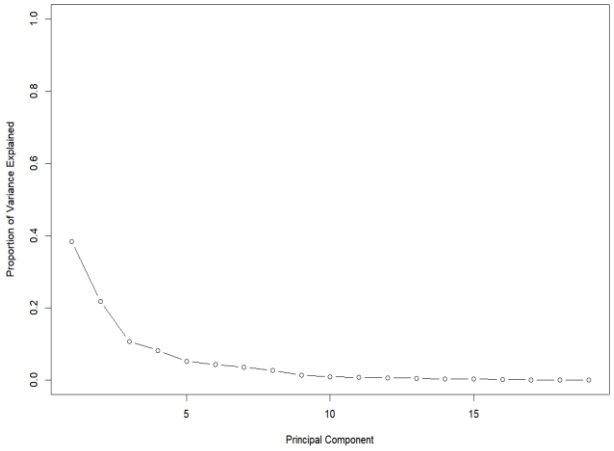
Problem 1-(a)

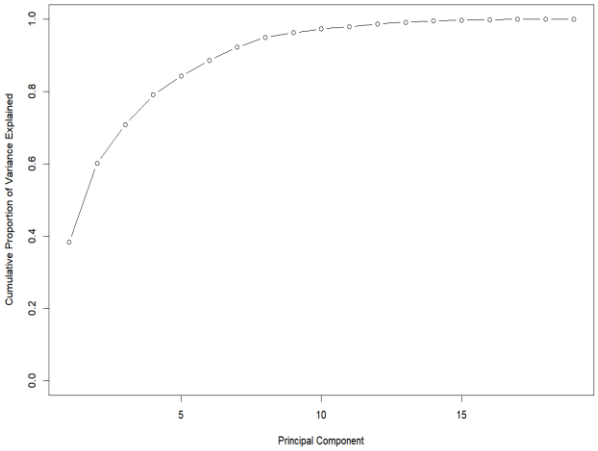
- If we failed to scale the variables before performing PCA, then most of the principal components that we observed would be driven by the CAtBat variable, since it has by far the largest mean and variance. Thus, it is important to standardize the variables to have mean zero and standard deviation one before performing PCA.

Problem 1-(b)

- I performed a PCA of the data with standardized variables. I made the table for PVE and Cumsum(PVE) according to the principal components, and I made the plot with this value. I set the data variance's as 90% to determine the number of PCs. So, I would recommend the number of PCs as 7 because the plot and the Cumsum(PVE) shows that 7 components results in variance more then 90%.

PC	PVE	Cumsum(PVE)
1	3.831424e-01	0.3831424
2	2.184108e-01	0.6015532
3	1.068636e-01	0.7084167
4	8.192520e-02	0.7903419
5	5.256081e-02	0.8429028
6	4.344504e-02	0.8863478
7	3.628108e-02	0.9226289
8	2.700156e-02	0.9496304
9	1.319648e-02	0.9628269
10	9.727217e-03	0.9725541
11	7.223413e-03	0.9797775
12	6.709461e-03	0.9864870
13	5.030866e-03	0.9915179
14	3.212465e-03	0.9947303
15	2.735578e-03	0.9974659
16	1.473967e-03	0.9989399
17	7.417156e-04	0.9996816
18	2.559159e-04	0.9999375
19	6.248919e-05	1.0000000

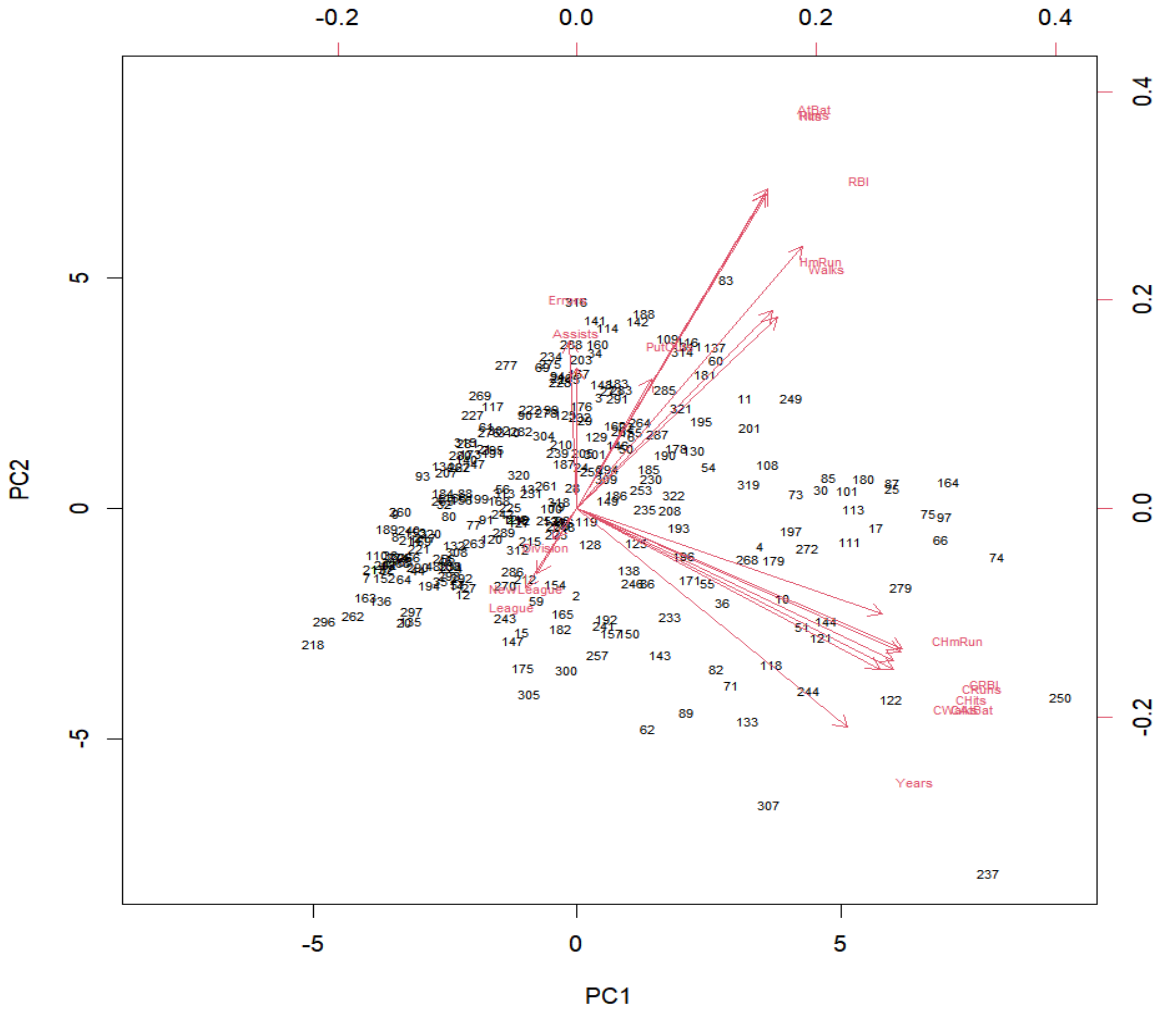




Problem 1-(c)

- I focused on the first two PCs and made the table showing correlations of the standardized quantitative variables with the two components (PC1 and PC2). I made the biplots showing the scores on the two components and the loadings on them. The Years, CatBat, Chit,..., Cwalks have positive loadings on first principal component, and AtBat Hits, HmRun,...,Walks have positive loading on the second principal component. So, the first components primarily measure in 1986 season, and the second components primarily measure the player's career.

	PC1	PC2
AtBat	0.1982903511	0.38378403
Hits	0.1958612933	0.37727112
HmRun	0.2043689229	0.23713561
Runs	0.1983370917	0.37772134
RBI	0.2351738026	0.31453120
Walks	0.2089237517	0.22960610
Years	0.2825754503	-0.26240195
CAtBat	0.3304629263	-0.19290382
CHits	0.3307416802	-0.18289883
CHmRun	0.3189794925	-0.12629732
CRuns	0.3382078595	-0.17227611
CRBI	0.3403428387	-0.16809208
CWalks	0.3168029362	-0.19231496
PutOuts	0.0776971752	0.15573663
Assists	-0.0008416413	0.16865189
Errors	-0.0078593695	0.20075992



Problem 2-(a)

- A good way to handle the problem is to standardize the data so that all standardized variables are given a mean of zero and a standard deviation of one. Then all variables will be on a comparable scale. If we don't standardize variables, it is likely to be biased when clustering. This is because groups in cluster analysis are defined based on the distance between points in mathematical space.

Problem 2-(b)

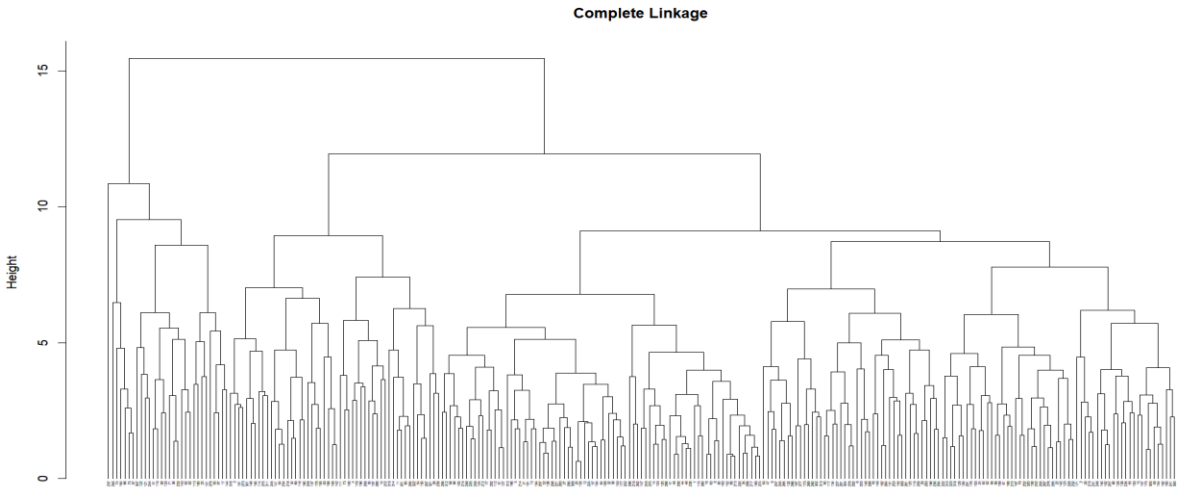
- I would use metric-based due to its effectiveness with numeric data.

Problem 2-(c)

- I did standardized the variables and operated hierarchically cluster the players using complete linkage and Euclidean distance with dist() function. The result of the dendogram below. And I cut the dendeogram at a height that results in two distince clusters (0 or 1). The summary of the cluster-specific means of the variables, which are standardized, is below table, and the mean salaries of the players when the cluster 1 is -0.1240844 and the cluster 2 is 0.9637218. When comparing C1

and C2, C1 has 233 observations and C2 has 30 observations. I could interpret this data that C2 tends to receive much more salary because their skills (such as AtBat, Hits,,,) are better on 1986 season and the players' career and they do less error when compared to C1 players.

	C1	C2
AtBat	-0.008615973	0.06691739
Hits	-0.004958349	0.03850985
HmRun	-0.062441905	0.48496546
Runs	-0.010526791	0.08175808
RBI	-0.062911598	0.48861341
Walks	-0.052877808	0.41068431
Years	-0.228886672	1.77768649
CAtBat	-0.257667855	2.00122034
CHits	-0.256546898	1.99251424
CHmRun	-0.264609651	2.05513496
CRuns	-0.258773795	2.00980980
CRBI	-0.280357634	2.17744429
CWalks	-0.258805538	2.01005635
League	-0.033085413	0.25696338
Division	-0.006125301	0.04757317
PutOuts	-0.029293620	0.22751378
Assists	0.043717417	-0.33953860
Errors	0.023253386	-0.18060130
NewLeague	-0.026487943	0.20572302



Problem 2-(d)

- I performed K-means with K=2 to cluster the players on the basis of standardized variables and the Euclidian distance. The summary of the cluster-specific means of the variables, which are standardized, is below table, and the mean salaries of the players when the cluster 1 is -0.3710137and the cluster 2 is 0.947589. When comparing C1 and C2, C1 has 189 observations and C2 has 74 observations. There is not a big difference when comparing to hierarchically cluster. So, I could interpret this data that C2 tends to receive much more salary because their skills (such as AtBat, Hits,,,) are good on 1986 season and the players' career and they do less error when compared to C1 players.

	C1	C2
AtBat	-0.1772004973	0.4525795
Hits	-0.1848657	0.4721570
HmRun	-0.2520322	0.6437040
Runs	-0.2040287	0.5211004
RBI	-0.2577735	0.6583674
Walks	-0.2374244	0.6063948
Years	-0.4193494	1.0710409
CAtBat	-0.4747936	1.2126484
CHits	-0.4743151	1.2114263
CHmRun	-0.4614815	1.1786488
CRuns	-0.4877376	1.2457082
CRBI	-0.4918975	1.2563329
CWalks	-0.4720683	1.2056879
League	0.1152035	-0.2942360
Division	0.08137728	-0.20784196
PutOuts	-0.08891692	0.22709862
Assists	0.04483091	-0.11450056
Errors	0.04876779	-0.12455557
NewLeague	0.08817909	-0.22521416

Problem 2-(e)

- I could recommend K-means with K=2. This is because there is not a big difference about the cluster-specific means of the variables and response between the hierarchically cluster and K-means cluster. However, the K-means cluster has twice players than hierarchically cluster. So, I think the K-means cluster well clustered rather than the hierarchically cluster. So, I could conclude that K-means cluster gives more sensible results.

Problem 3-(a)

- I fitted a linear regression model with "Hitters" data, and I got the test MSE as 0.4214063 when I performed leave-one-out-cross-validation to compute the estimated test MSE.

Problem 3-(b)

- By using PCR model with M as 16 comps chosen optimally via LOOCV, and the test MSE (with LOOCV), by using this M, is 0.4104077.

Problem 3-(c)

- By using PLS model with M as 12 comps chosen optimally via LOOCV, and the test MSE (with LOOCV), by using this M, is 0.4134874.

Problem 3-(d)

- By using the ridge regression with penalty parameter chosen optimally via LOOCV, the best lambda is 0.05336699, and the test MSE by using this lambda is 0.4082689.

Problem 3-(e)

- By performing linear regression, PCR model, PLS model, and ridge regression, I could get the test MSE 0.4214063, 0.4104077, 0.4134874, and 0.4082689 respectively. There are not big differences among them, but the ridge regression has the least test MSE. So, among these models, I would recommend the ridge regression.

```
library("pls")
library("glmnet")
library("bestglm")

setwd("C:/Users/haeun/OneDrive/문서/STAT33550")

#Problem 1
#Bringing the wine dataset
hitters <- read.csv("Hitters.csv")
hitters <- na.omit(hitters)
#Deleting the X variables
hitters$X <- NULL
hitters$Salary <- NULL

# Using dummy representation for them.
hitters$League <- ifelse(hitters$League == "N", 1, 0)
hitters$Division <- ifelse(hitters$Division == "W", 1, 0)
hitters$NewLeague <- ifelse(hitters$NewLeague == "N", 1, 0)
str(hitters)

#Problem 1-(a)
#Standardizing the variables
standarize_x <- scale(hitters)
apply(hitters, 2, mean)
apply(hitters, 2, sd)

#Problem 1-(b)
#Performing PCA with the standardized variables
pca <- prcomp(standarize_x, center = T, scale = T)
names(pca)

#Calculate the PVE and cumsum(PVE)
pc.var <- pca$sdev^2
pve <- pc.var/sum(pc.var)
cumsum(pve)
#Making the plot for the PVE and cumsum(PVE)
plot(pve, xlab = "Principal Component", ylab = "Proportion of Variance Explained", ylim = c(0,1), type = 'b')
plot(cumsum(pve), xlab = "Principal Component", ylab = "Cumulative Proportion of Variance Explained", ylim = c(0,1), type = 'b')

#Problem 1-(c)
pca$rotation[,1]
pca$rotation[,2]
biplot(pca, scale=0, cex = 0.55, expand = 0.95)

#Problem 2
hitters <- read.csv("Hitters.csv")
hitters <- na.omit(hitters)
#Deleting the X variables
hitters$X <- NULL
y <- scale(hitters$Salary)
hitters$Salary <- NULL
hitters$League <- ifelse(hitters$League == "N", 1, 0)
hitters$Division <- ifelse(hitters$Division == "W", 1, 0)
```

```
hitters$NewLeague <- ifelse(hitters$NewLeague == "N", 1, 0)

str(hitters)

#Problem 2-(c)
#Standardize the variables
standarize_x <- scale(hitters)
# Performing hierarchically cluster using complete linkage and Euclidean distance
hc.complete <- hclust(dist(standarize_x), method = "complete")

# Making the dendogram
plot(hc.complete, main = "Complete Linkage", xlab = "", sub = "", cex = 0.3, hang = -1)

# Cut the dendogram at a height that result in two distinct clusters
cuttree <- cutree(hc.complete, 2)
# Figuring out the index of cluster-specific
cuttree_1 <- which(cuttree == 1)
cuttree_2 <- which(cuttree == 2)
cuttree_1_variables <- standarize_x[which(cuttree == 1), ]
# The mean of the variables standardized with cluster 1
apply(cuttree_1_variables, 2, mean)
cuttree_2_variables <- standarize_x[which(cuttree == 2), ]
# The mean of the variables standardized with cluster 2
apply(cuttree_2_variables, 2, mean)

#Reload the data to get the Y(salary)
hitters <- read.csv("Hitters.csv")
hitters <- na.omit(hitters)
#Deleting the X variables
hitters$X <- NULL
y <- scale(hitters$Salary)
cuttree_1_y <- y[cuttree_1]
# The mean of the response standardized with cluster 1
mean(cuttree_1_y)
# The mean of the response standardized with cluster 2
cuttree_2_y <- y[cuttree_2]
mean(cuttree_2_y)

#Problem 2-(d)
set.seed(2)
#
km.out <- kmeans(standarize_x, 2, nstart = 20)
km.out$centers
km.out$cluster
index_1 <- which(km.out$cluster==1)
km.out.y1 <- y[index_1]
mean(km.out.y1)
index_2 <- which(km.out$cluster==2)
km.out.y2 <- y[index_2]
mean(km.out.y2)
plot(standarize_x, col = (km.out$cluster + 1), main = "K-Means Clustering Results with K=2", xlab = "", ylab = "", pch = 20, cex = 2)

#Problem 3
```

```
setwd("C:/Users/haeun/OneDrive/문서/STAT33550")
#Bringing the wine dataset
hitters <- read.csv("Hitters.csv")
hitters <- na.omit(hitters)
#Deleting the X variables
hitters$X <- NULL
#Factoring Leagut, Division, NewLeague
hitters$League <- as.factor(hitters$League)
hitters$Division <- as.factor(hitters$Division)
hitters$NewLeague <- as.factor(hitters$NewLeague)
# log(salary) dus to skewness in Salary
hitters$Salary = log(hitters$Salary)

#Problem 3-(a)

set.seed(2)
#Performing the linear regression model
linear <- lm(Salary~., data = hitters)
#compute the estimated test MSE by using LOOCV
error <- rep(1:nrow(hitters))
for(i in 1:nrow(hitters)){
  linear <- lm(Salary ~ ., data = hitters[-i, ])
  one_predic <- predict(linear, newdata = hitters[i,])
  error[i] <- (hitters$Salary[i]-one_predic)^2
}
mean(error)

#Problem 3-(b)
set.seed(2)
#Performing PCR model with LOOCV
pcr.fit <- pcr(Salary ~ ., data = hitters, scale = TRUE, validation = "CV", segments = nrow(hitters))
#### Look at results
summary(pcr.fit)
#Finding M chosen optimally via LOOCV
MSEP(pcr.fit)
sqrt(MSEP(pcr.fit)$val[1, 1,])
which.min(MSEP(pcr.fit)$val[1, 1,])
#Compute the test MSE of the model
error <- rep(1:nrow(hitters))
for(i in 1:nrow(hitters)){
  pcr.fit <- pcr(Salary ~ ., data = hitters[-i,], scale = TRUE, validation = "LOO")
  one_predic <- predict(pcr.fit, newdata = hitters[i,], ncomp = 16)
  error[i] <- (hitters$Salary[i]-one_predic)^2
}
mean(error)

#Problem 3-(c)
set.seed(2)
#Performing the PLS model with LOOCV
plsr.fit <- plsr(Salary ~ ., data = hitters, scale = TRUE, validation = "CV", segments = nrow(hitters))
#### Look at results
summary(plsr.fit)
#Finding the M chosen optimally via LOOCV
```

```
MSEP(plsr.fit)
sqrt(MSEP(plsr.fit)$val[1, 1,])
which.min(MSEP(plsr.fit)$val[1, 1,])
#Compute the test MSE of this model
error <- rep(1:nrow(hitters))
for(i in 1:nrow(hitters)){
  plsr.fit <- plsr(Salary ~ ., data = hitters[-i,], scale = TRUE, validation = "LOO")
  one_predic <- predict(plsr.fit, newdata = hitters[i,], ncomp = 12)
  error[i] <- (hitters$Salary[i]-one_predic)^2
}
mean(error)

#Problem 3-(d)
# Y is the Salary from the wine data set
y <- hitters$Salary
# X has all the variables except for the Salary, which is response
x <- model.matrix(Salary ~ ., hitters)[, -1]
# Making the grid and use it as lamba for glmnet function(ridge regression model)
grid = 10^seq(10, -2, length = 100)
ridge_mod <- glmnet(x, y, alpha = 0, lambda = grid)
# Setting the seed
set.seed(2)
# by using cross-validation with nfold(due to LOOCV) figuring out the best lambda
cv.out <- cv.glmnet(x, y, alpha = 0, nfolds = nrow(hitters), lambda = grid, grouped=FALSE)
bestlam <- cv.out$lambda.min
#Best Lambda is  0.05336699
#Compute the test MSE of this model
min(cv.out$cvm)
```